The Free/Libre/Open Source Software Ecosystem - How you can participate

Bruno Souza and Fabio Kon
Open Source Initiative - Education Working Group

(We encourage you to modify, adapt, extend, refine, and improve this set of slides and give your own customized presentation with it; you should put your name in the first slide and just add a small reference to our names in the last slide)

Agenda

- History
- Why FLOSS
- FLOSS organizations
- Open Source Definition
- Licenses
- Communities
  - Simon's onion
  - major repositories, foundations, numbers
  - events
- How you can participate
  - benefits, why participate, ethics, expected behavior, different kinds, "business" models

History - the beginnings

- 1950s and 1960s: software was almost always distributed with its source with little restrictions.

- 1970s: companies started to close their source and treat code as "industrial secret"

- On the other hand
  - 1973: Unix BSD is born
  - 1975: Ingres is born (later PostrgreSQL)
  - 1978: Knuth starts to work on TeX

The Free Software Movement

- Richard Stallman, MIT AI Lab, observes a shift from a free UNIX culture to a proprietary software culture in his working environment.
- 1983/4: Stallman leaves MIT and creates the GNU Project, proposing the creation of a new Unix implementation
  - emacs, X Window, GCC, TeX, ...
- 1985: Free Software Foundation
- 1987: FSF sells copies of GNU for USD 150 in magnetic tapes
- 1989: Copyleft and the GPL

The Free Software Definition

The four freedoms of software users:

0. Run the program for any purpose

1. Study how the program works, and change it. Access to the source code is a precondition for this.

2. Redistribute copies so you can help your neighbor.

3. Distribute copies of your modified versions to others. You give the whole community a chance to benefit from your changes.

Modern times

1991: Linus Torvalds makes his OS available

1992: GNU/Linux is born

- Some of the first FLOSS-based companies are born:
  - 1989: Cygnus (to support GNU products)
  - 1993: SuSE (slackware distribution, enterprise linux)
  - 1995: Red Hat
- 1995: MySQL
- 1996: KDE: better desktop interface

Open Source - a new complementary terminology for the same movement

1997: Eric Raymond. "The Cathedral and the Bazaar"
1997: GNOME
1998: Netscape opens its Mozilla browser
1998: Open Source Initiative (OSI) is founded
1999: Dell, HP, and SGI announce support for GNU/Linux
1999: Apache Foundation formed
2000: Sun opens StarOffice, creating OpenOffice.org
2001: IBM announces USD 1bi investment on Linux;
Wikipedia is created
2002: Creative Commons
2003: Motorola releases first cell phone with Linux
2004: First version of Ubuntu

Current times...

2005: ODF is recognized as OASIS standard, later ISO

2006: Sun opens Java Virtual Machine (OpenJDK)

2009: Oracle buys Sun

- OpenOffice donated to Apache, LibreOffice branched
- more investments in open source Java

Last 10 years: Great impact of FLOSS in

- Industry (now mainstream as part of the IT infrastructure for Internet, cloud computing, servers, tools)
- governments (Brazil, Germany, USA, Malasia)
- and society (helping drive freedom activism, promote knowledge sharing)

Why FLOSS

- Software can be seen as
  - knowledge
  - product
  - tool / platform

- FLOSS can maximize the benefits to society

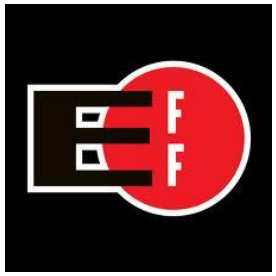
- FLOSS as an ethical option
  - common good, sharing, collaboration
- Enabler of the society of the future
  - closed software brings problems to governments, companies and citizens
- Platform for business
  - advantages for consumers and producers

Major Institutions

Advocate:

Other FLOSS:

Related:



Foundations

- Mozilla
  - Firefox, HTML5, tools
- Apache
  - HTTPD, Hadoop
- Linux
- Eclipse

mozilla
FOUNDATION

Apache

Open Source Definition (1/2)

According to the OSI, open source doesn't mean just access to code. An open source license must comply with:

1. Free Redistribution
2. Source Code
3. Derived Works
4. Integrity of the Author's Source Code
5. No Discrimination Against Persons or Groups

Open Source Definition (2/2)

An open source license must comply with:

6. No Discrimination Against Fields of Endeavor
7. Distribution of License
8. License Must Not Be Specific to a Product
9. License Must Not Restrict Other Software
10. License Must Be Technology-Neutral

Licenses are relevant

- Just making your code available on the Web doesn't make it open source.
- Copyright law restricts a lot what could be done with it (in most countries)

- You must clearly assign an open source license to it
  - specify the license in the source code and also make it explicit in the download page.
- Pick an OSI-approved, well-known license (don't reinvent the wheel).

3 types of licenses

1. Reciprocal
   - if you change the code and redistribute it, you must also redistribute the source code; the code will remain open source.
   - all the code linked to the code with a reciprocal license must remain with the same reciprocal license.

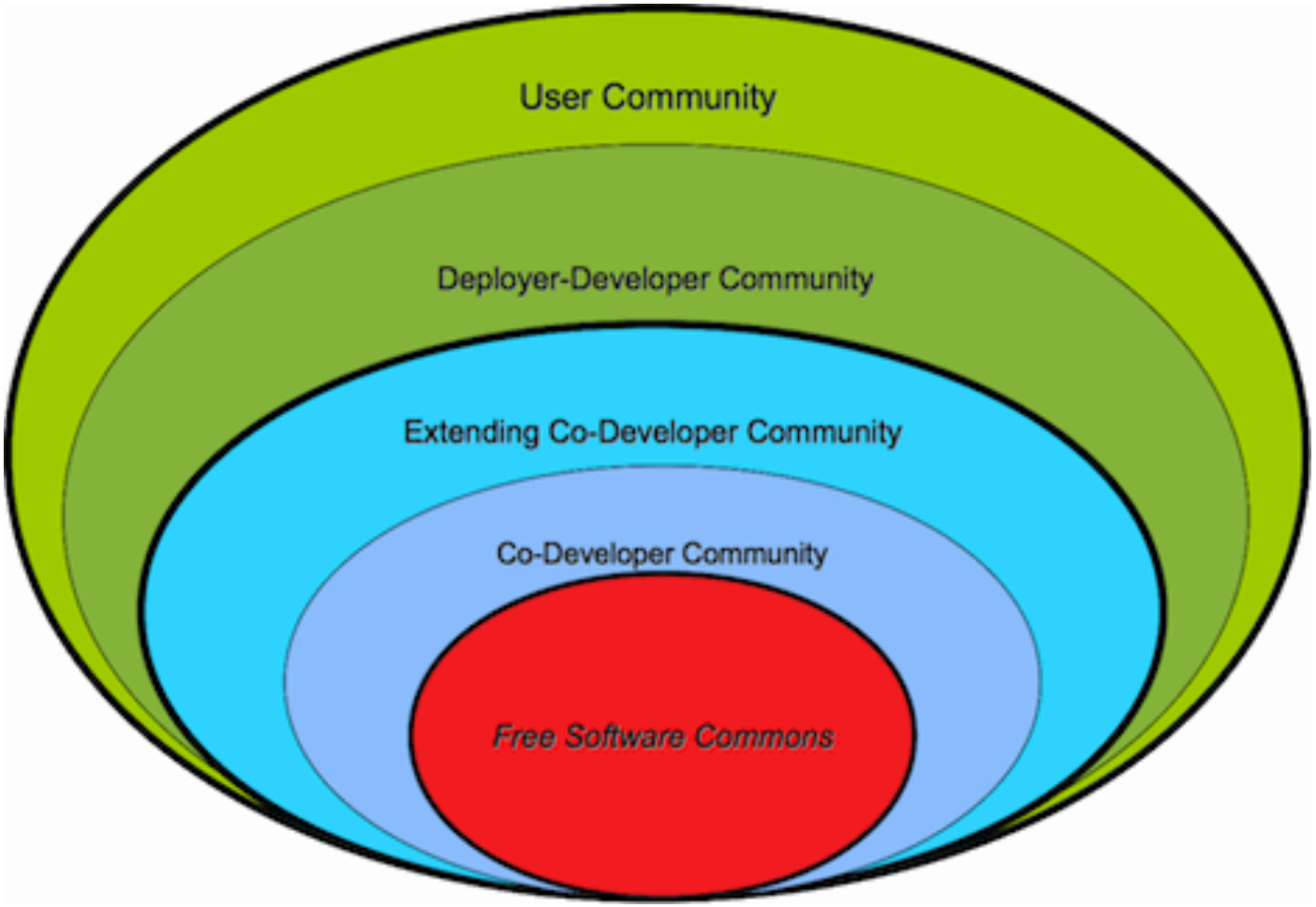
2. Partially reciprocal (file-based, weak copyleft)
   - similar to the reciprocal but you can distribute a single component of your code with this license and link it to code with other license (even proprietary).

3. Academic
   - you may relicense your derivative work under any license of your choice, or even make it proprietary.

Popular and widely used licenses

- Academic
  - Apache License 2.0
  - BSD
  - MIT license
- Reciprocal
  - GNU General Public License (GPL)
- Partial, file-based reciprocal, weak copyleft
  - Eclipse Public License (EPL)
  - GNU Lesser General Public License (LGPL)
  - Mozilla Public License 2.0 (MPL)

Open Source Communities



Repositories

GitHub

Savannah

Find a project that interest YOU!

Many, many developers!

- SourceForge: 316,624 projects
- GitHub: 3,012,331 repositories, 1,067,856 users
- Java.net: 2,132 projects, 716,303 users
- Apache: 100+ projects, 2000+ committers
- RubyForge: 9,281 projects, 92,701 users
- Savannah: 3,391 projects, 53,966 users
- Launchpad: 24,997 projects, 1,796,156 translations
- Codeplex: 25,064 projects

Events

- Meet people and learn how to participate
- Look for an event near your home
- Follow online or go to the main ones
  - US: OSCON
  - Europe: FOSDEM
  - Latin America: FISL

How can you participate?

- Every good work of software starts by scratching a developer's personal itch.
  - Eric Raymond - The Cathedral and the Bazaar
- Work on something you need, chances are, every other developer needs it too!
- Fix a bug that annoys you.
- Write or improve a documentation that you can't find.
- Solve a problem that you have.

How can you participate?

- To solve an interesting problem, start by finding a problem that is interesting to you.
  - Eric Raymond - The Cathedral and the Bazaar
- Find something you want to do, that gets you excited
- Nothing beats doing what you like
- Search the repositories for projects that match what you want
- Search for projects that take you where you want to go
- Yes, it is all about YOU!

How can you participate?

- If you have the right attitude, interesting problems will find you.
  - Eric Raymond - The Cathedral and the Bazaar
- Once you realize that this is for YOUr benefit...
- Once you understand open source is good for YOU...
- Once you get involved because YOU want...
- Once you understand how much this will improve YOUr abilities...

How can you participate?

- Good programmers know what to write. Great ones know what to rewrite (and reuse).
  - Eric Raymond - The Cathedral and the Bazaar
- Open source is about the source code
  - is learning from others
  - is learning how to build on top of what others did
  - is learning how to let others build on what you did!
- Open source is not about licenses...

open source is all about being

a great developer!

Small, practical things you can do...

- Use open source software
  - you can't learn something you don't use!
- Once you find a problem, report it
  - this helps you learn how the project works
  - try to find on the source code why the problem exist
- Read lots of source code
  - This is Rule #1 of being a rockstar developer!
  - only open source really gives you source code...
- Join the mailing list and answer questions
  - write documentation, a presentation, a blog
  - teaching others helps you learn more than anything

Spread the word, improve the world!

- Promote open source to developers
  - it is the natural way for developers to learn
  - make better developers, make better code
- Promote open source ideas in other areas
  - social activism
  - privacy protection
  - e-gov and open data
  - piracy vs copyright discussions
  - knowledge sharing
- Open source is the way to learn...
  - and also the way to teach and help!

The future starts now...

Open Source is a mainstream way of developing software, that reaches millions of developers across the globe.

You too can participate.

Choose your way, start now!

www.opensource.org

<name_of_the_presenter>

<email_of_the_presenter>

References

OSI: opensource.org

Open source definition: opensource.org/osd.html

FSF: www.gnu.org/philosophy/free-sw.html